



AMENDMENTS TO THE SPECIFICATION

Please amend the Specification as follows:

Please replace the paragraphs numbered [0009] - [0043] with the following paragraphs [0009] - [0043.1]:

[0009] Aspects of the present invention may be found in a method for updating an electronic device. The method may comprise updating a first code version in the electronic device to a second code version using update information comprising a set of instructions for converting the first code version to the second code version, and converting data associated with the first code version to a form compatible with an the second code version.

[0010] In an embodiment of the present invention, the method may further comprise retrieving a list of names of converter utilities associated with the update information.

[0011] In an embodiment of the present invention, the method may further comprise storing update information in at least a portion of memory in the electronic device.

[0012] In an embodiment of the present invention, the method may further comprise communicating queries regarding availability of update information from the electronic device to a server.

[0013] In an embodiment of the present invention, the method may further comprise retrieving the update information and a list of names of converter utilities in a single retrieval operation.

[0014] In an embodiment of the present invention, the first code version may comprise a plurality of software applications, and each of the plurality of software applications may be associated with a corresponding converter utility.

[0015] In an embodiment of the present invention, the update information may be capable of updating the plurality of software applications in a single update event.

[0016] In an embodiment of the present invention, the method may further comprise generating an update package reference, the update package reference may at least comprise an update package location memory reference and a list of names of converter utilities memory reference.

[0017] In an embodiment of the present invention, the method may further comprise determining whether a code update is necessary, and if it is determined that an update is not necessary, then performing a reboot operation.

[0018] In an embodiment of the present invention, the method may further comprise determining whether a code update is necessary. If it is determined that an update is necessary, then retrieving data from an update package reference, verifying authenticity of the update information, updating the first code version, executing at least one converter utility associated with at least one software application, communicating an update confirmation to at least one external system, and performing a reboot operation.

[0019] In an embodiment of the present invention, executing at least one converter utility associated with the at least one software application may comprise retrieving and updating associated security information.

[0020] In an embodiment of the present invention, retrieving and updating associated security information may comprise retrieving an authorization related object and associating it with the at least one software application.

[0021] In an embodiment of the present invention, the method may further comprise retrieving a list of security information associated with the update information, and installing the security information after updating the first code version.

[0022] In an embodiment of the present invention, executing at least one converter utility associated with the at least one software application may comprise converting security information comprising authentication and authorization information.

[0023] In an embodiment of the present invention, executing at least one converter utility associated with the at least one software application may comprise converting subscription information.

[0024] In an embodiment of the present invention, the first code version may comprise a plurality of software applications. The at least one converter utility may comprise a plurality of converter utilities. Each converter utility may be associated with a corresponding application software application.

[0025] In an embodiment of the present invention, each of the software applications for which an update was determined to be necessary may be updated in a single update event.

[0026] In an embodiment of the present invention, executing at least one converter utility associated with the at least one software application may comprise converting data associated with each software application in a single conversion event.

[0026.1] In an embodiment of the present invention, code may comprise firmware.

[0027] Another aspect of the present invention may be found in a machine-readable storage, having stored thereon a computer program having a plurality of code sections. The code sections may be executable by a machine for causing the machine to perform operations for updating an electronic device. Such an embodiment may comprise code comprising an update agent, for coordinating updating of code in the electronic device using update information comprising a set of instructions for converting the code to an updated code, and code comprising a converter, for invoking a converter utility associated with the code and for converting data associated with the code to a form compatible with an updated version of the code.

[0028] In an embodiment of the present invention, the electronic device may comprise a communication layer for communicating a list of names of converter utilities associated with the update information and communicating queries regarding availability of update information from the electronic device to a server.

[0029] In an embodiment of the present invention, the communication layer may be adapted to communicate the update information and the list of names of converter utilities in a single communication event.

[0030] In an embodiment of the present invention, the electronic device may comprise memory for storing update information.

[0031] In an embodiment of the present invention, the code may comprise a plurality of software applications and each of the software applications may be associated with a corresponding converter utility.

[0032] In an embodiment of the present invention, the update information may be adapted to update the plurality of software applications in a single update event.

[0033] In an embodiment of the present invention, the machine-readable storage may comprise a placement layout table for mapping a memory location of update information. The placement layout table may at least map an update information memory location and a list of names of converter utility memory location.

[0034] In an embodiment of the present invention, the machine-readable storage may comprise means for determining whether an update of code is necessary. If it is determined that an update is not necessary, then the electronic device performs a reboot operation.

[0035] In an embodiment of the present invention, the machine-readable storage may also comprise means for determining whether an update of the code is necessary. If it is determined that an update is necessary, then the update agent may retrieve data from an update package reference, verify authenticity of the update information, and update the code. The converter may execute at least one converter utility associated with the code. A communication layer may communicate an update confirmation to at least one external system, and the electronic device may perform a reboot operation.

[0036] In an embodiment of the present invention, the code may comprise a plurality of software applications. The at least one converter utility may comprise a plurality of converter utilities. Each converter utility may be associated with a corresponding software application.

[0037] In an embodiment of the present invention, the update agent may be adapted to update each of the software applications for which an update was determined to be necessary in a single update event.

[0038] In an embodiment of the present invention, the converter may execute at least one converter utility associated with each software application and may convert data associated with each of the software applications in a single conversion event.

[0039] In an embodiment of the present invention, the converter may convert security data associated with the code. The converted security data may enable access to the updated version of the code.

[0040] In an embodiment of the present invention, the electronic device may comprise a communication layer. The communication layer may communicate converter utilities. The converter utilities may be adapted to converting security data and access control information associated with the update information. The communication layer may also communicate information associated with the converted security data and access control information from the electronic device to a server.

[0041] In an embodiment of the present invention, a converter utility may invoke downloading of update information and firmware/software update information.

[0042] In an embodiment of the present invention, security information may be updated after a software/firmware update has been performed.

[0043] In an embodiment of the present invention, a software application may be updated when data associated with the software application changes.

[0043.1] In an embodiment of the present invention, code may comprise firmware.

Please amend paragraph [0048] on page 9, as follows:

[0048] Fig. 1 illustrates a block diagram of an update network 105 for updating an electronic device according to an embodiment of the present invention. Mobile handset 107 is one example of an electronic device in which an embodiment of the present invention may be practiced. Mobile handset 107 may employ firmware/software update system 111 for updating the mobile handset firmware/software based upon information contained in at least one update package retrieved from server 109. In a representative embodiment of the present invention, an update package (that may be referred to herein as update information) may comprise a plurality of transformation instructions which transform a first code version into a second code version. In one aspect, the instruction set utilizes a conversion process employed by a client device such as, for example, the mobile handset 107 to efficiently convert an existing the first code version to the second code version. The mobile handset 107 may be communicatively coupled to server 109 via communication link 143.